Acta Astronautica 94 (2014) 395–408

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Design and development of guidance navigation and control algorithms for spacecraft rendezvous and docking experimentation

Giorgio Guglieri [a,]*, Franco Maroglio [b], Pasquale Pellegrino [c], Liliana Torre [d]

[a] *Politecnico di Torino, Dipartimento di Ingegneria Meccanica e Aerospaziale, Associate Professor, Corso Duca degli Abruzzi 24, Torino, Italy*

[b] *TESEO S.p.a., Technical Manager, Corso Alexander Fleming 27, Druento, Italy*

[c] *Thales Alenia Space, Domain Exploration & Science Italy, Avionic Systems & Robotics, Head of Robotics & Mechatronics Group, Strada Antica di Collegno 253, Torino, Italy*

[d] *Politecnico di Torino, Dipartimento di Ingegneria Meccanica e Aerospaziale, currently ALTRAN Italia, Strada del Drosso 33/19 Pal C, Torino, Italy*

ARTICLE INFO

Article history:
Received 19 May 2012
Received in revised form
7 February 2013
Accepted 18 February 2013
Available online 5 March 2013

Keywords:
Guidance navigation control
Rendezvous docking
Ground testing facilities
Docking mechanisms
Data fusion and digital filtering

ABSTRACT

This paper presents the design of the GNC system of a ground test-bed for spacecraft rendezvous and docking experiments. The test-bed is developed within the STEPS project (Systems and Technologies for Space Exploration). The facility consists of a flat floor and two scaled vehicles, one active chaser and one "semi-active" target. Rendezvous and docking maneuvers are performed floating on the plane with pierced plates as lifting systems. The system is designed to work both with inertial and non-inertial reference frame, receiving signals from navigation sensors as: accelerometers, gyroscopes, laser meter, radio finder and video camera, and combining them with a digital filter. A Proportional-Integrative-Derivative control law and Pulse Width Modulators are used to command the cold gas thrusters of the chaser, and to follow an assigned trajectory with its specified velocity profile. The design and development of the guidance, navigation and control system and its architecture—including the software algorithms—are detailed in the paper, presenting a performance analysis based on a simulated environment. A complete description of the integrated subsystems is also presented.

1. Introduction

The development of Rendezvous and Docking space systems [1] requires an experimental analysis [2] in order to test and validate the docking technologies, the control systems, the navigation methods, the proximity operations [3], the model algorithms and the numerical simulations. Nevertheless, costs and time prevents the possibility of the on-orbit experimental approach [4], so the availability of ground-based test facilities that can reproduce the conditions of a RV&D maneuver is strongly desirable [5]. The scenario, regarding this type of test-beds, involves different research activities and developments, focused on the implementation, validation and verification of the guidance, navigation and control algorithms. Some examples are shortly presented hereafter, without claiming to be exhaustive.

At the US Naval Postgraduate School in Monterey, three generations of in-plane orbital RV&D test-beds have been realized [6], the last of which has demonstrated both the validity of navigation filters with GPS, magnetometer and gyroscope sensors and collision avoidance

* Corresponding author. Tel.: +39 011 0906860.
E-mail address: giorgio.guglieri@polito.it (G. Guglieri).

http://dx.doi.org/10.1016/j.actaastro.2013.02.010

algorithms. The MIT SPHERES [7] (Synchronized Position Hold Engage and Reorient Experimental Satellite) project uses the NASA's 2D laboratory platform and the International Space Station as test-bed to validate control algorithms, autonomous docking [8] and high-risk measurements between three cooperative vehicles. The EPOS (European Proximity Operation Simulator) facility, recently modernized, is another example of hardware-in-the-loop simulator offering the capability of real-time, real-size simulation, combined with realistic simulation of Sun in the sensor's field of view. Verification of GNC algorithms and data processing is possible thanks to the Motion System Control linked to the simulator [9]. Another relevant example is the Technion's Distributed Space Systems Laboratory, developed for formation flight ground experiments [10]. Finally, at NASA Jet Propulsion Laboratory, it was demonstrated the value of a formation flight algorithm named FAST (Formation Algorithms and Simulation Test-bed), conceived to bring out issues in data fusion, inter-spacecraft communication and sensing. The aim of this algorithm is to find, through the use of reaction wheels, gyros, star trackers and a ground hardware formation test-bed, Earth-like planets orbiting distant stars [11].

2. Present work

The present research is carried out in the frame of STEPS program–Systems and Technologies for Space Exploration—a research project co-financed by Regione Piemonte (Piedmont Region) within the P.O.R.-F.E.S.R. 2007–2013 EC program. One of the aims of this program is to reproduce on the ground the phases of a spacecraft RV&D mission starting from the phasing up to the docking. Considering the vehicles operating in an orbital scenario (such as those involving the International Space Station) and the same time scale, linear velocities and distances are scaled. Then the test-bed represents the scaled condition, with a scale factor of 1/18, of two vehicles that are getting out of the inertial navigation and entering in the far range rendezvous phase, characterized by a relative navigation system. Once the chaser probe is inside the target cone, the docking phase starts allowing the docking mechanism final latching.

The developed guidance, navigation and control algorithms shall support the vehicles (3-DOFs) docking with the required precision. The GNC shall also interface with the MMA (Mission Management Agent) software (that detects possible failures or mission violation), the sensors and the actuators. Only the active vehicle, the chaser, is linked with the ground Control Center (CC) in order to exchange wireless telemetry and command data. The passive vehicle, the target, does not communicate neither with the Control Center nor with the chaser, having a predefined trajectory.

Main contributions of this work to the state of the art are:

- Development of a technological area (laboratory) with hardware equipment (docking mechanism, navigation sensors, communication layer) and software functions (GNC, mission planning and graphical user interfacing);
- Development and evaluation of various types of navigation filters able to manage information from multiple sensors simultaneously, allowing a redundant sensing configuration, including the handling of failures;

The paper is organized as follows: the first section presents the spacecraft simulator used for the experiments. The second section outlines the GNC architecture, including the navigation filter in its "information" form, the control logics that actuate thrusters for maneuvering and the guidance algorithm. The third section describes the "plant subsystem" that is: the reference equations of motion, the model of sensors and thrusters including actuation errors. Last section shows the simulation results in order to demonstrate the potential controllability of the vehicles and to show the effect of a sensor failure on the overall performance.

3. The experimental rig

The experimental rig allows for the verification of guidance, navigation and control algorithms in an in-plane close proximity flight condition. The vehicles float using a thin air film of 60 μm over a very flat floor, reproducing a nearly frictionless environment in planar dynamic with 3-DOFs, i.e. two degrees of freedom for translation and one for rotation. The docking mechanism function has been preliminarily tested; the aim of the docking mechanism is connecting the two vehicles by means of a probe and three latches, docking the cone, hosted by the target vehicle (Fig. 1).

The target vehicle is supposed to keep on moving on a pre-definite trajectory (known by the chaser) while the chaser vehicle has full maneuver capabilities. Vehicles motion is controlled by 14 cold gas actuators with 3 different levels of thrust.

The vehicles (Fig. 2) consist of one cylindrical air tank placed vertically on a lifting plate and two decks. On the decks take place all the electric and electronic devices together with the thrusters, the pneumatic system and the docking mechanism. There are three different reference frames: a fixed reference coinciding with the floor ($O_{iner}: x_{iner}, y_{iner}$), a body reference for each vehicle ($O_{chaser}: x_{chaser}, y_{chaser}$) and a relative reference frame centered in the target center of gravity ($O_{rel}: x_{rel}, y_{rel}$) (Fig. 3). The docking mechanism and the thrusters are mounted on the middle deck, which is closer to the vehicle center of gravity, avoiding the generation of torque disturbances along x-body and y-body reference frame.

The chaser is equipped with 4 sensors: radio finders, laser position finders, video camera (providing position and attitude measurements with respect to the target vehicle) and an Inertial Measurement Unit (IMU) providing linear accelerations and angular velocity measurements with respect to the chaser-body reference frame. Body acceleration and angular velocity are then bias-compensated and transported in the inertial reference frame ($O_{iner}: x_{iner}, y_{iner}$). The laser position finder can perform measurements with respect to the inertial reference

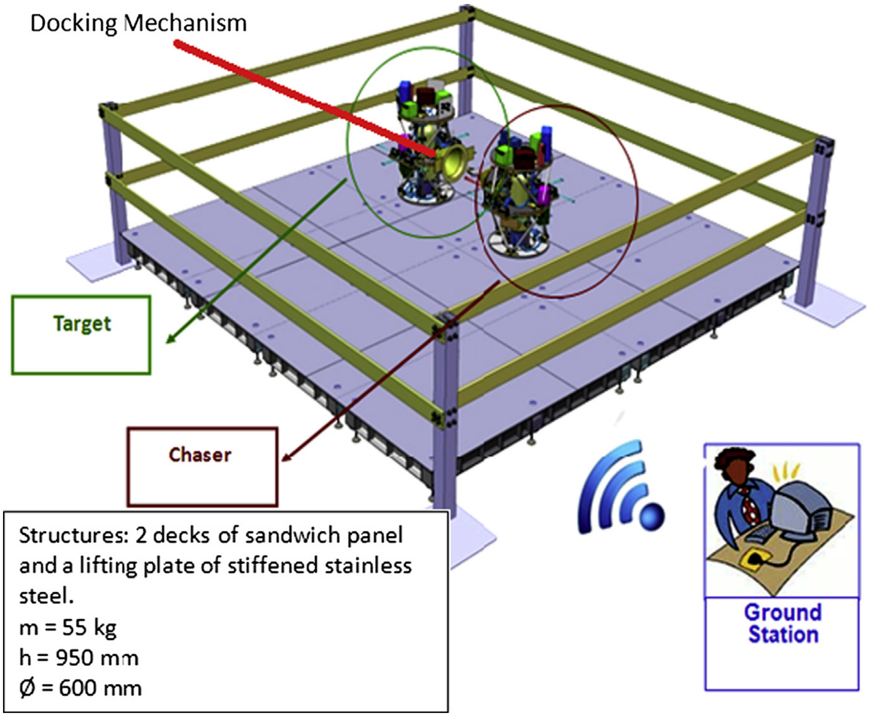

Docking Mechanism
Target
Chaser
Structures: 2 decks of sandwich panel and a lifting plate of stiffened stainless steel.
m = 55 kg
h = 950 mm
Ø = 600 mm
Ground Station

Fig. 1. The experimental rig: the chaser (right) and the target (left).

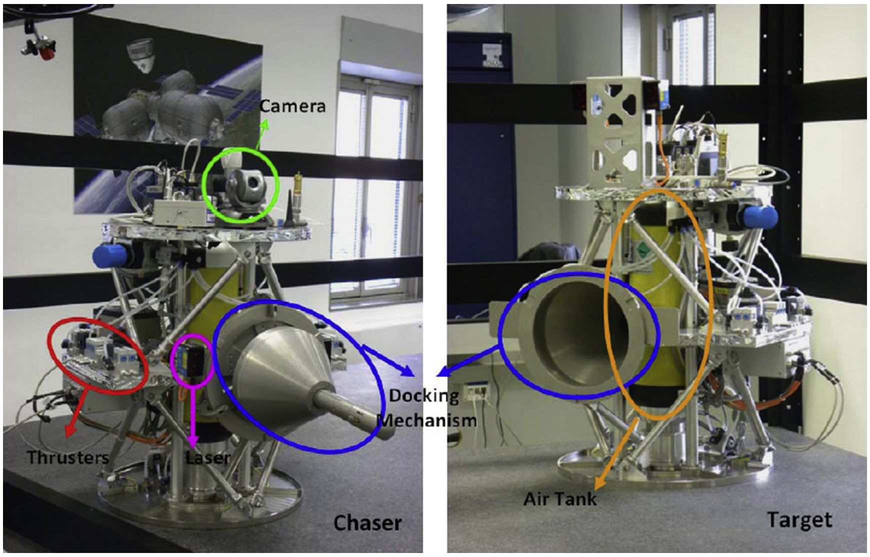


Fig. 2. The RV&D experimental rig: chaser and target vehicles.

frame and it's used also to correct IMU's drift. The sensor application range is outlined in Fig. 4, where also the required precisions are indicated, as a function of relative distance. These precisions have been calculated applying the scaling factor approach defined previously. Up to 2 m of relative distance the absolute navigation is used. During relative navigation (below 2 m) the linear precision required is 5.5 mm and the angular one is 1°.

Unlike the chaser, the target doesn't need a relative navigation system; it carries the transmitter of the radio finder system and the optical marker for the camera, required by the chaser as reference for its navigation sensors. The target moves with respect to the inertial reference frame system by using only an IMU and the laser position finders.

The GNC software has been developed in Matlab-Simulink® environment. Executable C-files are then created via Simulink Coder® tool (for "nidll.tlc" target) and uploaded on the processor of chaser and target vehicles. The GNC system communicates with other two software tools: the Control Center (CC) and the MMA supervisor module. The CC, running on the remote control station PCs, collects all the information provided by the entire system via wireless communication (i.e. expected and estimated states, thruster command, mission phase, recovery state, feed line pressure, lifting and thrusters air flow), making them available for the users with a graphical interface developed in LabView® environment. The user can define a new mission through the CC, which sends to chaser GNC the parameters initialization. The MMA software takes care of the planning, execution of the maneuvers and data exchange with the CC (failure detection, isolation/identification and recovery modules). This module interacts continuously with the GNC system: every time an alarm is detected, a command of station keeping is sent to the GNC as recovery action. The MMA compares the estimated state and the expected state from GNC, checking that the maximum allowable errors are not exceeded. The MMA also compares the information from the sensors in order to verify their consistency. In both cases an alarm signal is detected, then a management by exception logics is implemented.

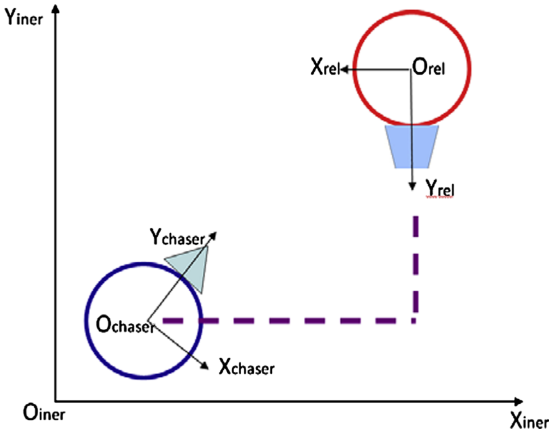


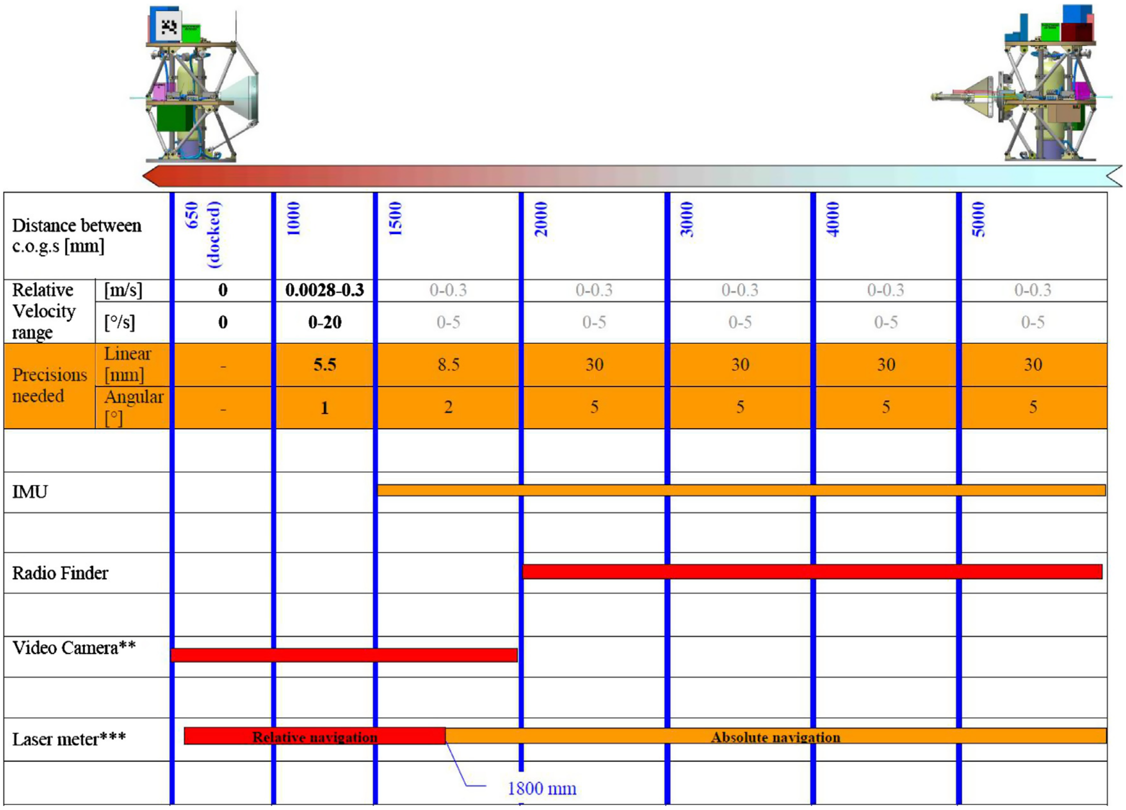
Fig. 3. Reference frames used in the test-bed.

Figs. 5 and 6 give an overall view of the complete configuration of the chaser and target subsystems. The Table 1 summarizes the main electronic hardware and drivers of the vehicles.

4. The GNC algorithm

The GNC algorithm has to translate the instructions defined in a Mission Plan Matrix (MPM) into vehicles' desired position and attitude (Guidance), to estimate the vehicles' position according to sensors information (Navigation) and issue the opening/closing command to the thrusters (Control). Currently the algorithms of the chaser's GNC allow to:

1) Move chaser vehicle along the path predefined in the Mission Plan Matrix, calculating its own position with respect to both the target and the ring reference frame.



Distance between c.o.g.s [mm]		650 (docked)	1000	1500	2000	3000	4000	5000
Relative Velocity range	[m/s]	0	0.0028-0.3	0-0.3	0-0.3	0-0.3	0-0.3	0-0.3
	[°/s]	0	0-20	0-5	0-5	0-5	0-5	0-5
Precisions needed	Linear [mm]	-	5.5	8.5	30	30	30	30
	Angular [°]	-	1	2	5	5	5	5

Fig. 4. Sensor operative range and precision requirements.

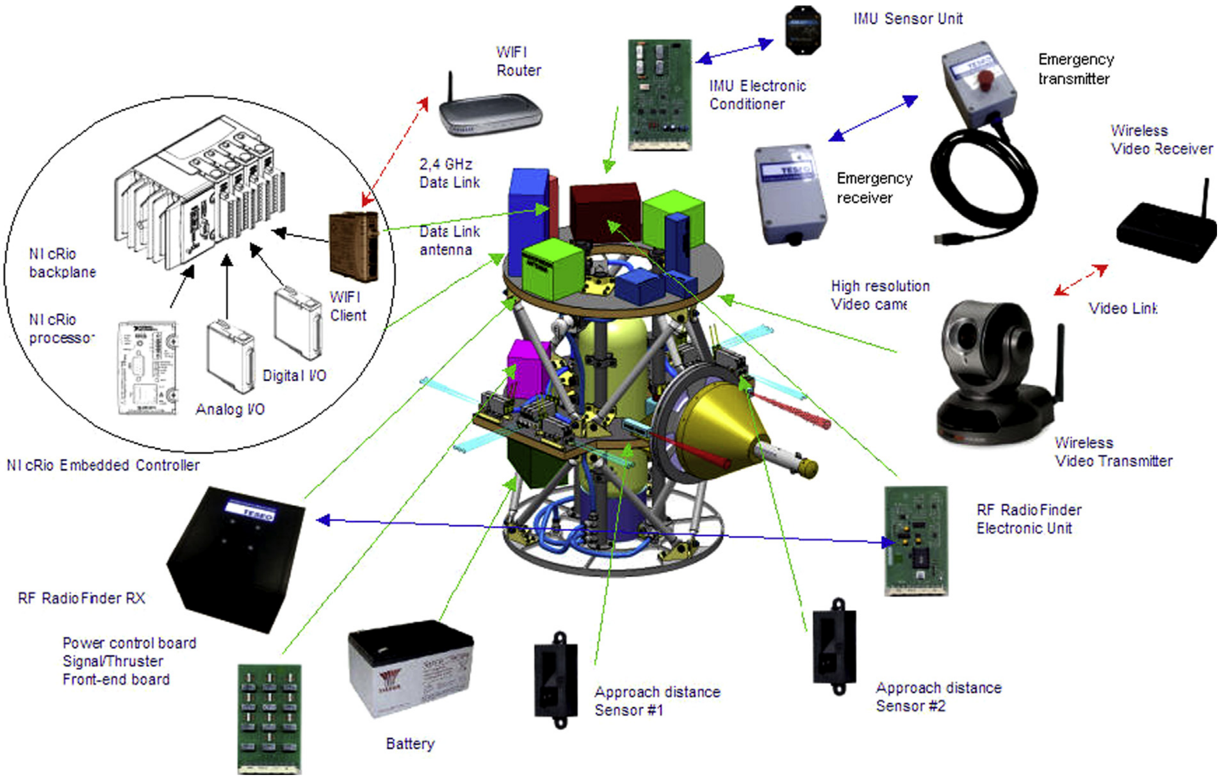

IMU Sensor Unit
WIFI Router
IMU Electronic Conditioner
Emergency transmitter
Wireless Video Receiver
2,4 GHz Data Link
Emergency receiver
Data Link antenna
NI cRio backplane
NI cRio processor
WIFI Client
High resolution Video came
Video Link
Digital I/O
Analog I/O
Wireless Video Transmitter
NI cRio Embedded Controller
RF RadioFinder Electronic Unit
RF RadioFinder RX
Power control board Signal/Thruster Front-end board
Battery
Approach distance Sensor #1
Approach distance Sensor #2

Fig. 5. Chaser equipment.

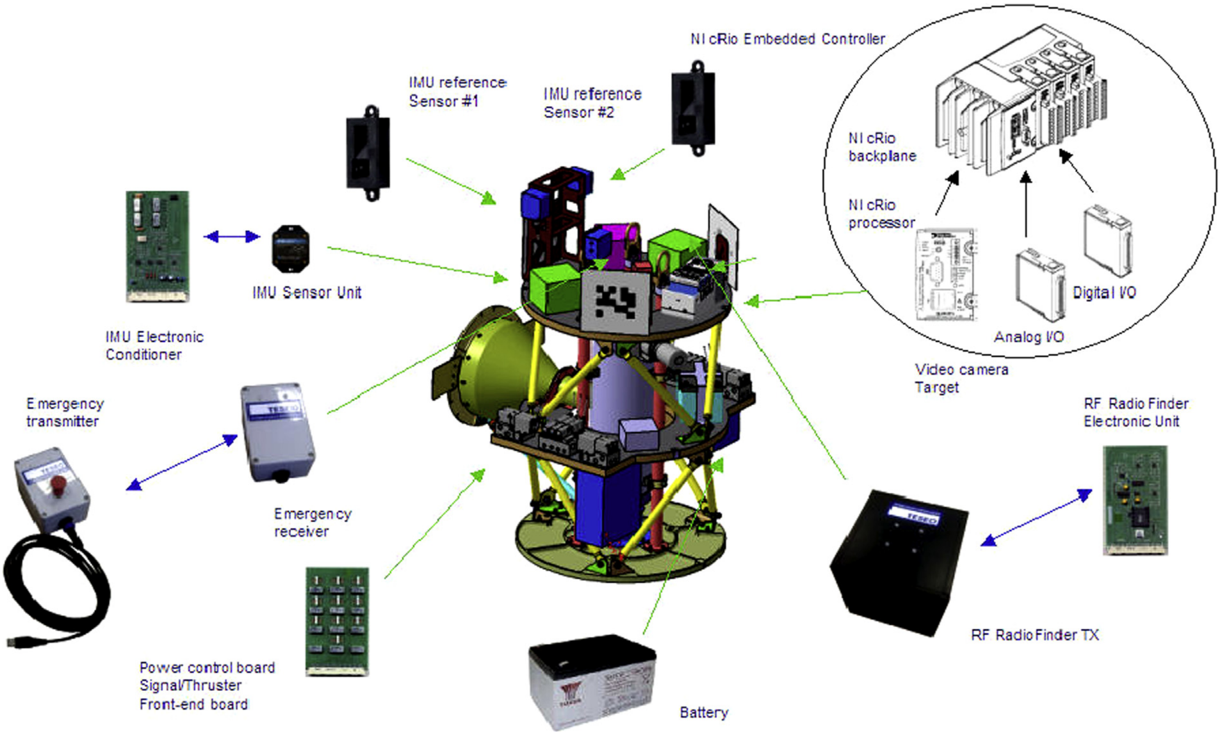


Fig. 6. Target equipment.

Table 1
Electronics hardware and sensors description.

Part's name description	Details	Description	Power consumption (W)
Main Control Unit	Embedded FPGA type VIRTEX-5 LX30	Embedded block RAM: 1,152 Kbits. Processor: MPC5200 Free scale 400 MHz Lab View® Real Time Operating System.	8 W
Peripheral Devices		WI-FI Client IEEE802.11.i full compatibility. Supported protocols: ARP, UDP, TCP, Telnet, ICMP, SNMP, DHCP, HTTP, SMTP. 32 Channels Digital I/O, TTL Level with independent channels I/O configuration. 32 Channels Analog input, ±200 mV to ± 10 V input range, 16 Bit resolution.	7 W
WI-FI Router	Installed on the chaser, allows to communicate with Control Centre	IEEE802.11.i full compatibility. TKIP encryption. 64/128 bit WEP encoding.	3 W
Laser Position Sensor	Triangulation mode distance acquisition referenced to a test-bed peripheral ring and distance acquisition referenced to the target mirrors	Measurement rate: 0.02 s Distance measurement calibration curve for Accuracy: ±0.2%, Repeatability: 0.5% Measurements independent from surface characteristics or colors	3 × 1,5 W
Radio Finder Sensor	Based on a triaxial isotropic sensors array (installed on the chaser vehicle) able to receive and discriminate an encoded emission from an isotropic source (installed on the target vehicle).	Angular accuracy: ±2.5 ° Angular position repeatability: ±2° Position calibration functions Digital data output Minimum dynamic bandwidth: 100 Hz Signal/noise ratio: >30 dB	6 W
Video Camera Sensor	Installed on the chaser, recognize the marker on target Connected to the acquisition system through Wireless LAN data link.	DLINK DCS-6620G	8 W
IMU sensor	Installed on the chaser and the target, IMU is updated in position from the laser sensors absolute position data.	Output modes: 100 Hz for Euler angles MEMS Sensors suite: accelerometers, gyroscope, magnetometers, temperature 0.1% minimum Repeatability; 0.20° Accuracy: ±0.5° typical for static test conditions; ±2.0° typical for dynamic (cyclic) test conditions and for arbitrary orientation angles (no drift)	1 W
Power and Emergency Management Unit	Controls the system power distribution to the chaser and target vehicles.	Interface the Main Control Unit to the on-board devices (Thrusters, pneumatic valves, transducers) Generate, following an Operator action, a protected RF command able to shut-down the chaser or the target operations.	2 W

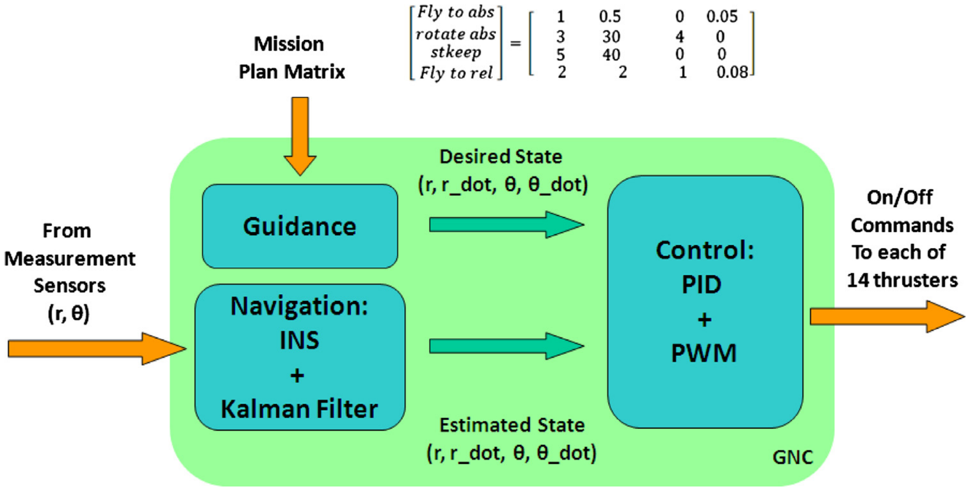


Fig. 7. GNC architecture.

2) Receive information from MMA about failure of sensors and about of new mission plan as a consequence of a corrective action.

3) Send to the Control Center the telemetry information and receive back data about the images gathered by the camera.

The control center receives the image via wireless link, it performs the image processing and it re-sends position and attitude to the chaser. The control center represents a remote auxiliary computation unit.

4.1. The guidance algorithm

The guidance algorithm is implemented to maneuver the chaser along the selected trajectory that can be defined in terms of waypoints (x_i,$y_i$or θ_j) and velocity (V_{di}or $\dot{\theta}_j$) listed in the Mission Plan Matrix (Fig. 7):

$$MissionPlanMatrix=\begin{bmatrix} i & x_i & y_i & V_{di} \\ i & x_i-x_{i-1} & y_i-y_{i-1} & V_{di} \\ j & \theta_j & \dot{\theta}_j & 0 \\ j & \theta_j-\theta_{j-1} & \dot{\theta}_j & 0 \\ i & T_{stk} & 0 & 0 \end{bmatrix} \tag{1}$$

The first column indicates the type of maneuver (i stands for translational or station keeping maneuver,j stands for attitude maneuver). The second and third columns indicate the waypoint coordinates to be reached, also expressed as function of preceding waypoints (x_i-x_{i-1},y_i-y_{i-1},$\theta_j-\theta_{j-1}$). In case of a station keeping maneuver, the second column T_{stk} indicates the duration, in seconds, of the maneuver itself. The desired state in terms of velocity and position profiles reflects the typical sequence acceleration-drift–deceleration both in translational and angular maneuvers between two waypoints. So the guidance algorithm calculates the time of each translational maneuver:

$$\Delta T_{lin}=\frac{\Delta S}{(1-k)V_{di}} \tag{2}$$

where $\Delta S=\sqrt{x_i^2+y_i^2}$is the total displacement read from MPM of the i-th maneuver, V_{di} is the drift velocity read from third column of MPM and k is the percent duration of the transient at constant acceleration with respect to the total maneuver time between two waypoints. In this way the velocity profile is a ramp signal with a constant acceleration equal to:

$$a_i=\frac{V_{di}}{k\Delta T_{lin}}=\frac{V_{di}^2(1-k)}{k\Delta S} \tag{3}$$

The present simulations are obtained assuming $k=0.1$. Similarly the attitude maneuver has a duration equal to:

$$\Delta T_{att}=\frac{\theta_j}{(1-k)\dot{\theta}_j} \tag{4}$$

that leads to a constant angular acceleration:

$$\ddot{\theta}_j=\frac{\dot{\theta}_j}{k\Delta T_{att}}=\frac{\dot{\theta}_j^2(1-k)}{k\theta_j} \tag{5}$$

In this way a sequence of linear and angular maneuvers, followed with constant acceleration are created as function of time. The desired state is now compared with the estimated state, computed by the navigation algorithm, and the difference is rotated in the body reference frame. This error is the input for the control algorithm. The architecture of GNC subsystems is represented in Fig. 7.

4.2. The navigation algorithm

The measurements from the different sensors are fused in a single estimate for position and velocity by an Information Kalman Filter (IKF) [12]. The IKF performs a data fusion process between all the measurements coming from the proper sensors. This particular form of the filter was chosen due to its simple additive nature of the update stage resulting computationally simpler than the equation of classic Kalman filter [13], at the cost of increased complexity in prediction.

The measurements available from the chaser's $i-$sensor are position and attitude, both in relative and absolute reference frame and they are represented by the vector $\overline{z}_i(k)=[xy\theta]$. Thanks to the presence of the IMU, there are two possible choices of the control vector u inside the filter: the first equals u to the acceleration and angular velocity from the IMU; the second equals u to the forces and torque resulting from the ON–OFF thrusters output signal. The first case prevents the use of INS integration's results inside the measurement vector $\overline{z}_i(k)$, so during absolute navigation if a failure of the laser meter occurs the mission should be interrupted, as the laser is the only measurement available. Instead the second case allows to use the INS integration inside the measurement vector $\overline{z}_i(k)$, since the control vector u is a derivation of command thruster actuation. So if the laser sensor fails during absolute navigation the redundancy is guaranteed by the INS. Since the accuracy of laser is 2 mm and the drift of INS navigation errors is the order of 5 cm every 40 s, the second navigation's architecture is chosen. Therefore the data from INS navigation were considered inside the measurement vector containing a proper noise and drift rate (directly reset by laser measurements). The time of INS reset is dictated by the requirement of accuracy on position and attitude.

The IKF equations of prediction stage for a system with discrete-time dynamic model are:

$$\overline{x}(k,k-1)=\overline{\overline{\Phi_k}}\overline{x}(k-1,k-1)+\overline{\overline{B_k}}\overline{\overline{u}}(k) \tag{6}$$

$$\overline{\overline{P}}(k,k-1)=\overline{\overline{\Phi}}_k\overline{\overline{P}}(k-1,k-1)\overline{\overline{\Phi_k^T}}+\overline{\overline{G_k}}\,\overline{\overline{Q_k}}\,\overline{\overline{G_k^T}} \tag{7}$$

where $\overline{x}=[x,y,\theta,\dot{x},\dot{y},\dot{\theta}]$ is the state vector of the system, $\overline{\overline{\Phi}}_k$is the state transition matrix, $\overline{\overline{B}}_k$is the input matrix and $\overline{\overline{u}}$ the control vector, $\overline{\overline{G_k}}=\overline{\overline{\Phi_k}}\,\overline{\overline{B_k}}$is the process noise coupling matrix. The error covariance matrix $\overline{\overline{P}}$is computed taking into account the system noise represented by the covariance matrix $\overline{\overline{Q}}_k$. The measurements vector is:

$$\overline{z_i}(k)=\overline{\overline{H_i}}\overline{x}(k,k-1)+\overline{b_i}(k,k-1)\ \ i=1,\ldots.N \tag{8}$$

where N is the number of sensors, $\overline{\overline{H_i}}$ represents the measurement matrix and $\overline{b_i}$ is the zero-mean white Gaussian (ZMWG) noise affecting the measurement with an assumed known covariance matrix $\overline{\overline{R_i}}(k)=E[\overline{b_i}(k)b_i^T(k)]$.

This last matrix changes its values depending on:

1) The relative distance between vehicles, in order to switch from absolute to relative measurements.

2) The alarm signal from MMA about a detected sensor failure. If the $i-$sensor is OFF or broken $\overline{b_i}(k)=100$ so that the information will not be reliable.

Given the Eqs. (6) and (8) it is possible to calculate the state estimation. The estimation stage is computed by:

$$\bar{\hat{x}}(k,k)=\bar{x}(k,k-1)+\sum_{i}^{N}\overline{\overline{K_i}}[z_i(k)-\overline{\overline{H_i}}\bar{\hat{x}}(k,k-1)] \quad i=1,....N \tag{9}$$

where the state $\bar{\hat{x}}$ is the state prediction and $\overline{\overline{K_i}}$ is the Kalman gain matrix which realizes the balancing between the sensors measurements and the dynamic state predictions. The error covariance $\overline{\overline{P}}(k)$ matrix is updated taking into account the Kalman gains calculated separately for each sensor.

$$\overline{\overline{P}}^{-1}(k,k)=\overline{\overline{P}}^{-1}(k,k-1)+\sum_{i}^{N}\overline{\overline{H_i}}^{T}\overline{\overline{R_i}}^{-1}\overline{\overline{H}}_i \quad i=1,....N \tag{10}$$

$$\overline{\overline{K}}_i=\overline{\overline{P}}(k,k-1)\overline{\overline{H_i}}^{T}\overline{\overline{R_i}}^{-1} \quad i=1,....N \tag{11}$$

The calculation of the state begins entering the prior estimate, $\bar{x}(0)$ and $\overline{\overline{P}}(0)$, then the Kalman gain matrix is computed with Eq. (11) and the prior estimate is updated with measurements by using Eqs. (9) and (10) This is the filtered state estimate to be used by the control system at time k, and the new prior estimate is given by the prediction of system dynamics for time $k+1$. Once the state is estimated, it is compared with the expected state coming out from Guidance algorithms.

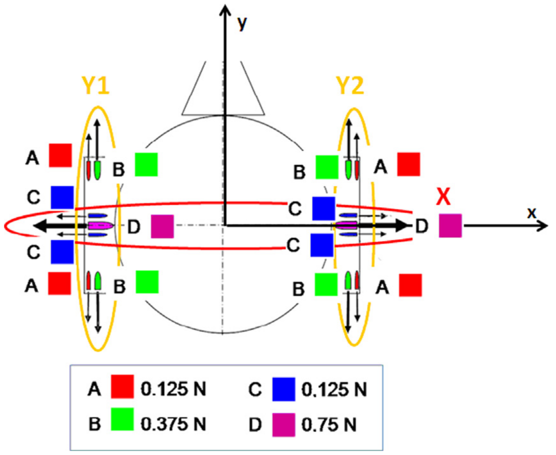


Fig. 8. Thrusters grouping.

Table 2
Possible thruster configuration.

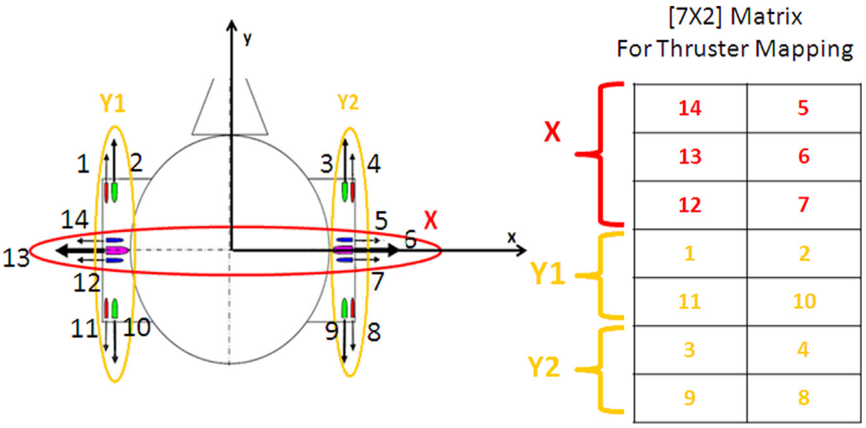
Level	Thrust [N]	Configuration	Level	Torque [Nm]	Configuration
F_1	0.25	A or C	M_1	0.0685	A
F_2	0.75	A and C	M_2	0.1912	B
F_3	1.00	A+B and C+D	M_3	0.529	A+B

4.3. The control algorithm

The Control algorithm receives as input (with a frequency of 2 Hz) the difference between desired and estimated state and then it gives as output (with a frequency of 2 Hz) the time opening command for the 14 electro valves. Then the PWM manages these opening timing at a frequency of 20 Hz. This algorithm works considering the body reference frame $(CoG,chaser: x_{body},y_{body},z_{body})$, where the second axis is oriented along the docking mechanism, z_{body} coincides with the axis of symmetry of the vehicles and it is oriented upwards, x_{body} completes the right-handed reference frame. In the control algorithm, the 14 cold thrusters of the propulsion system, are grouped into three separate sets as shown in Fig. 8:

- Group X: The thrusters C and D belong to this group and they allow translational motion along X-axis.
- Group $Y1$: The thrusters A and B with x-negative coordinates belong to this group.
- Group $Y2$: The thrusters A and B with x-positive coordinate belong to this group.

The Y-groups allow the translational and rotational motion if properly switched.



X	14	5
	13	6
	12	7
Y1	1	2
	11	10
Y2	3	4
	9	8

Fig. 9. Thrusters mapping.

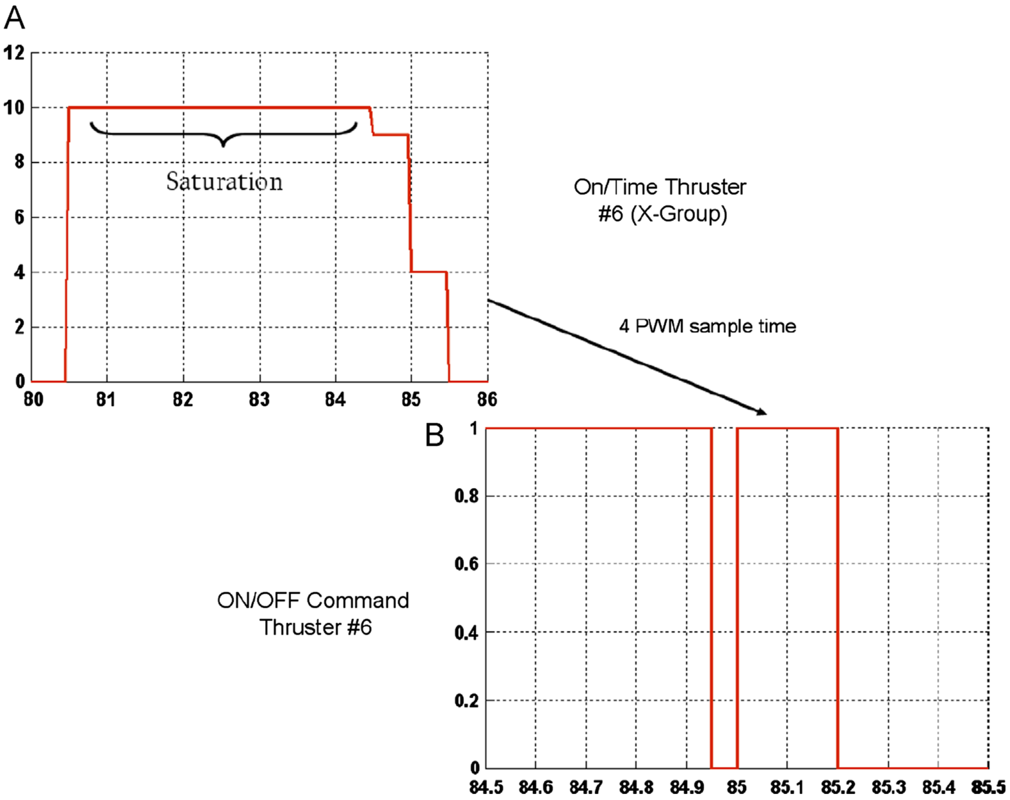


Fig. 10. (A) On-time during deceleration phase and (B) corresponding ON/OFF command.

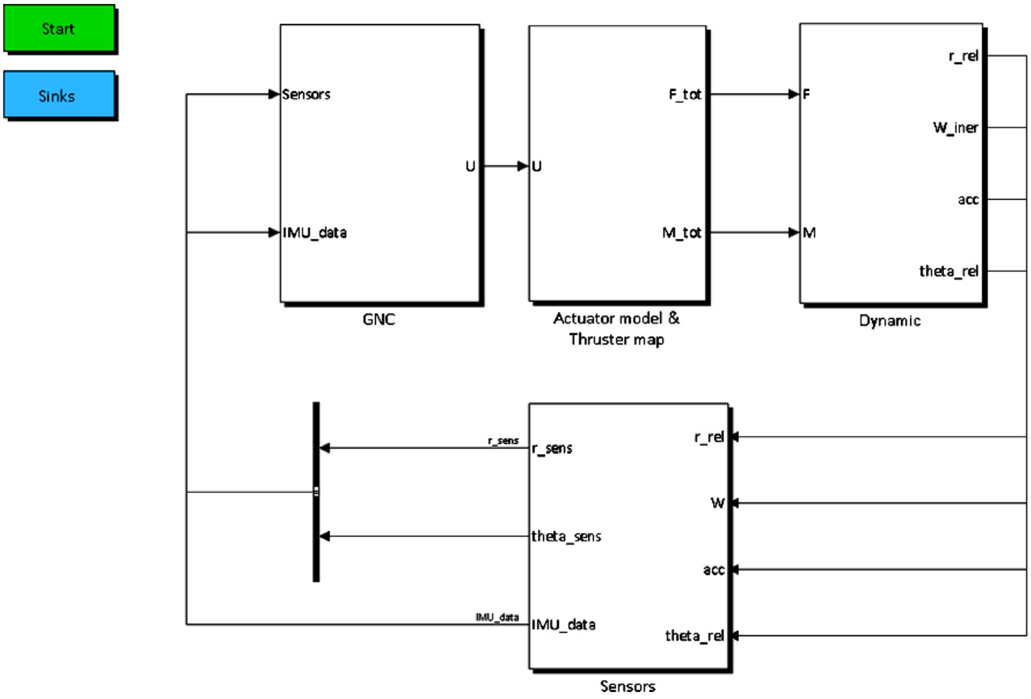


Fig. 11. The Simulink® GNC screenshot.

Each group has three levels of thrust and the selected level is automatically set depending on the impulse required to minimize the error along x_{body},y_{body} directions and θ_{body} attitude (difference between estimated and desired state). The data in Table 2 associate the three levels of thrust for the 14 cold gas thruster.

The impulse required is calculated using a PID control logic for each of the three DOFs. The control logic is activated only if the error on position and/or attitude and/or velocity is outside a dead band: the more stringent dead band is set during the last 1 m range, for the position is ± 5.5 mm, and that of attitude is $\pm 1^\circ$. If the error between desired and estimated state remains within these limits no command is produced. For example, along the y-axis (coinciding with Y-group) the required impulse is equally distributed between $Y1$ and $Y2$ groups and its value is:

$$I_y(t) = \Delta t_{control}[K_P(y_{nav}(t) - y_{gui}(t)) + K_D(\dot{y}_{nav}(t) - \dot{y}_{gui}(t)) + K_I(\int y_{nav}(t) - y_{gui}(t))]$$

In the case of x-axis the pulse is assigned to the X-group. The three gains K_p,K_d,$K_i$are calculated using the Ziegler–Nichols tuning method. In the equation $\Delta t_{control} = 0.5s$ is the time interval at which the control generates the parameter τ_{thr}, ranging from 0 to 10 (where 10 represents the saturation of signal), defined as:

$$\tau_{thr} = \frac{I_{Y1}}{F_{Y1\max}} \frac{1}{\Delta t_{PWM}} \tag{12}$$

where $I_{Y1} = I_y(t)/2$ is the required pulse for the $Y1$ group, $F_{Y1\max}$ is the maximum thrust deliverable for that level, among those listed in Table 2 and $\Delta t_{PWM} = 0.05s$ is the time interval at which a PWM control generates the ON/OFF commands for the electro valves. Thus τ_{thr} is initially calculated for X, $Y1$ and $Y2$ groups, and then referred to each thruster through a mapping matrix, described in Fig. 9.

At this point the PWM receives as input a $[7 \times 2]$ matrix containing the on-time from 0 to 10 of each thruster and compares it to a repeating sequence ramp. From this comparison the ON/OFF Boolean command signal for the single thruster (with a rate of 20 Hz) is calculated.

The evolution of τ_{thr} for the thruster no 6 during a maneuver and the corresponding Boolean command produced by the PWM are plotted in Fig. 10. If $\tau_{thr} = 5$ the PWM output is a ON-command of 0.025 s. The saturation is reached during acceleration and deceleration transients.

5. The plant

The plant is represented through the modeling of test-bed dynamics in Matlab-Simulink® environment, in which it is possible to test the behavior of the above-mentioned GNC system without the hardware in the loop.

The plant is composed of three main systems (Fig. 11) that emulate the real test-bed functions and so directly or indirectly connect the GNC system to:

- Actuator Model: it takes into account the ON/OFF thruster switching time and the uncertainty of the thrusters position on the vehicle.
- Dynamics Model: it's the 3 DOFs dynamic model of two rigid bodies with constant mass in relative motion.

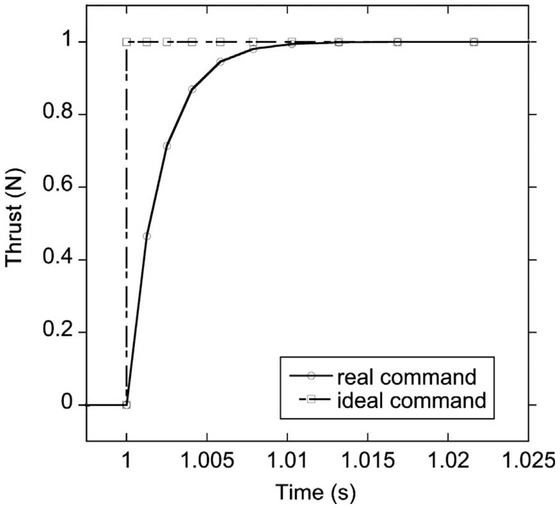
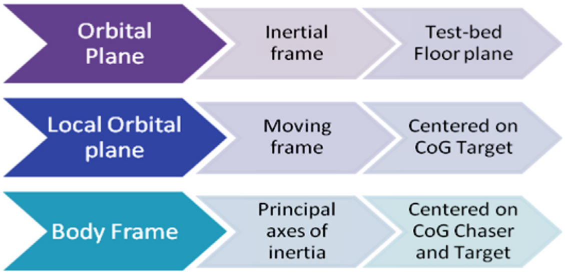


Fig. 12. Actuator thrust model (N).



Orbital Plane	Inertial frame	Test-bed Floor plane
Local Orbital plane	Moving frame	Centered on CoG Target
Body Frame	Principal axes of inertia	Centered on CoG Chaser and Target

Fig. 13. Orbital and test-bed reference frames.

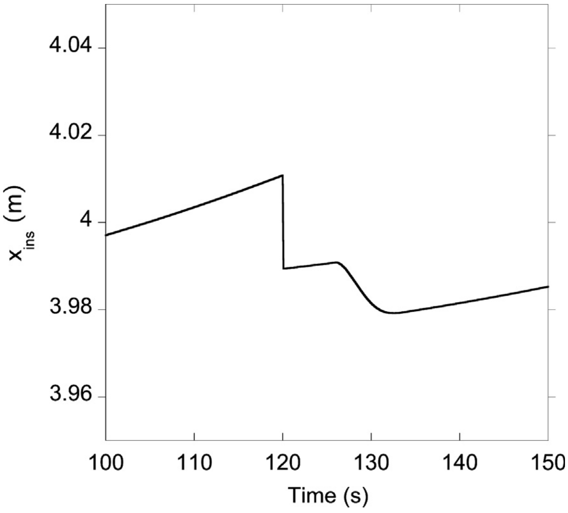


Fig. 14. Effect of laser reset on INS integration along x coordinate (m).

- Sensor Model: it implements the disturbances and the errors of the sensors.

5.1. The actuator model

The 14 thrusters are ideally fixed in their positions and perfectly distributed with respect to the center of mass, so there are no residual torque error due to misalignment. The Actuator Model allows calculation of force and torque applied to the vehicles that will be used by both the IKF, as control parameter u, and the Dynamic Model to integrate the motion and attitude equations.

The ON/OFF thrusters switching time is modeled applying a transfer function of the first order to the PWM output signal of each thruster:

$$F(s) = \frac{1}{ks+1} \tag{13}$$

since the ON commutation time is equal to 8 ms, the coefficient of S is $k = 0.008/n$ where k indicates that 98% of the thrust is reached after 8 ms. So the transfer function is $F(s) = 1/0.002s+1$. In Fig. 12 the ideal ON command is compared with the real one [14].

5.2. The Dynamics model

Since the chaser is meant to work both with the inertial and the relative reference frame, the dynamic model implemented computes the motion and attitude of the chaser either with respect to the test-bed floor plane and to the moving frame, centered into the center of gravity of the target vehicle. Fig. 13 shows the analogies with the docking orbital case.

Using the inertial reference frame (up to 2 m relative distance), chaser motions are described by the rigid body 3 DOFs dynamic equations with constant mass, and by the

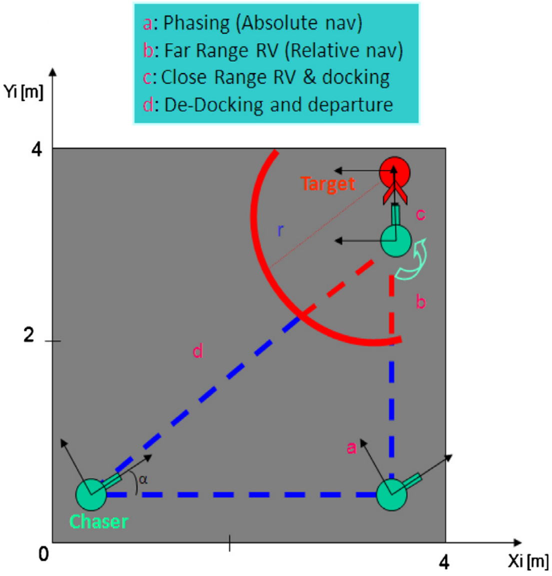

a: Phasing (Absolute nav)
b: Far Range RV (Relative nav)
c: Close Range RV & docking
d: De-Docking and departure
Yi [m]
4
2
0
4
Xi [m]
Target
r
c
b
d
a
α
Chaser

Fig. 15. Mission profile.

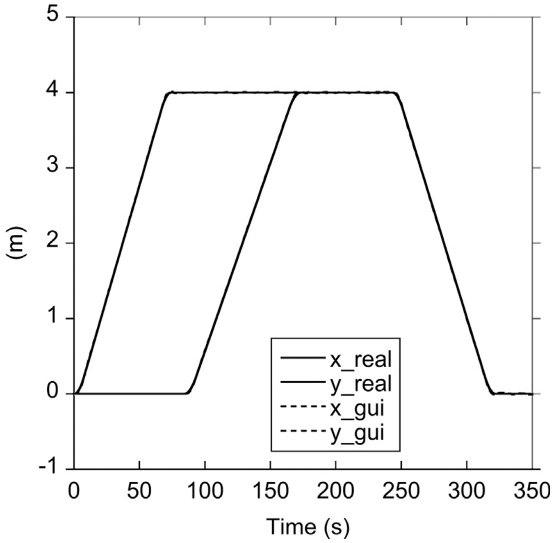


Fig. 16. Comparison between desired and simulated path (m).

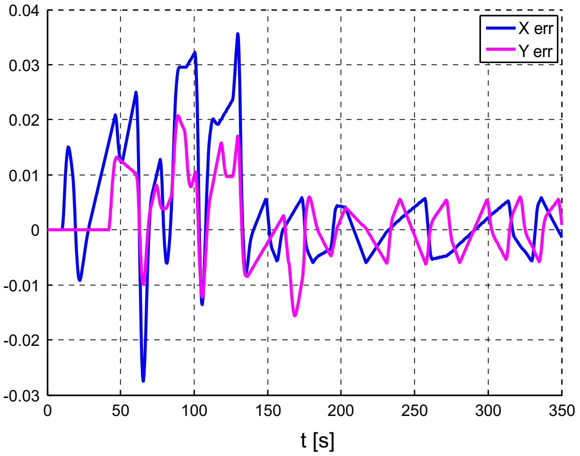


Fig. 17. Error between desired and simulated position (m).

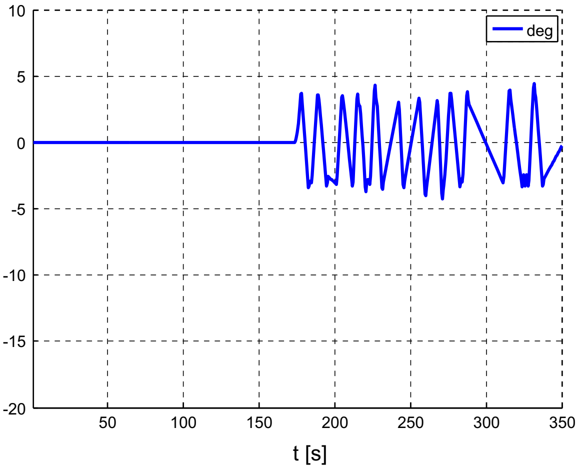


Fig. 18. Error between desired and simulated attitude (deg).

Euler equation for the attitude:

$$\vec{r} = \frac{\vec{F}_C}{m_C} \quad \ddot{\theta} = \frac{M_C}{I_C} \tag{14}$$

where m_c and I_C are respectively the mass and the inertia of the chaser. $\vec{F}_C$ and M_C are the force (along x, y) and the torque (around z) applied to the chaser, calculated through the Actuator Model.

Considering the relative reference frame, the chaser uses relative sensors information to estimate its position and attitude with respect to the target vehicle. The dynamic equations usually implemented in this case are the Clohessy-Whiltshire [15] linear equations written into local orbital plane centered on the CoG of the target. For chaser's maneuvers performed in the vicinity of the target, i.e. the local orbital plane origin, elapsing a short time in comparison to the orbital period, the equations can be further simplified in a double integrator for both x and y directions, that also reflects the dynamics of the vehicles of the test bed. So the equations implemented are (see Ref. 6 for details):

$$\ddot{x}_{rel} = \frac{F_{xT}}{m_T} - \frac{F_{xC}}{m_C} \tag{15}$$

$$\ddot{y}_{rel} = \frac{F_{yT}}{m_T} - \frac{F_{yC}}{m_C} \tag{16}$$

$$\ddot{\theta}_{rel} = \dot{\omega}_{zT} - \dot{\omega}_{zC} = \frac{M_T}{J_T} - \frac{M_C}{J_C} \tag{17}$$

where the subscript *rel* indicates the relative reference frame, centered in the target CoG.

As we can see from the equations above, the use of the relative system equations requires the knowledge of the target state. For this reason all the tests are performed with an assigned dynamics of the target that in this case can be expressed in terms of forces F_T and torque M_T applied to the target.

5.3. The Sensors model

From the calculation of position and attitude, thanks to the equations of motion, the sensor performance are modeled in order to reproduce the measurement vector z as input to the IKF. The laser, the camera and the radio-finder sampling frequencies are respectively 50 Hz, 10 Hz and 20 Hz and they are modeled as a sum of a zero mean Gaussian noise with assigned standard deviation plus a constant bias (see Table 1).

The performance of the IMU [16] takes into account for several types of errors, like:

- Bias repeatability: it's constant during functional operation but different at any power on of the instrument.
- Angle/velocity random walk, modeled as zero mean Gaussian noise at high frequency, and it's due to the thermo-mechanical noise of the instrument.
- Bias stability, due to flicker noise, and it is modeled as a random walk and it generates a drift factor up to 30 s of operation.

Measurements about angular rate and acceleration, detected by the z-gyroscope and the accelerometers are integrated by the INS producing position and attitude bias that will be reset by the laser every 40 s (Fig. 14).

6. Results of simulation

A typical mission profile with fixed target is shown in Fig. 15, in which the initial chaser position is (0,0) and the target is fixed at coordinates (4,4) with an attitude of -90°.

The MPM is:

$$MPM = \begin{bmatrix} 1 & 4 & 0 & 0.06 \\ 2 & 0 & 4 & 0.05 \\ 3 & 90 & 2 & 0 \\ 2 & -4 & -4 & 0.08 \end{bmatrix}$$

So the chaser switches from absolute frame to relative reference frame after 130 s at a relative distance from the target equal to 2 m. Fig. 16 compares the desired path with the simulation.

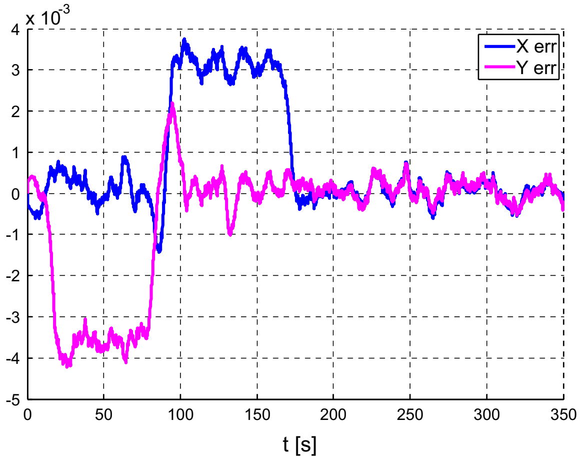


Fig. 19. Error between real and Kalman filter estimated position (m).

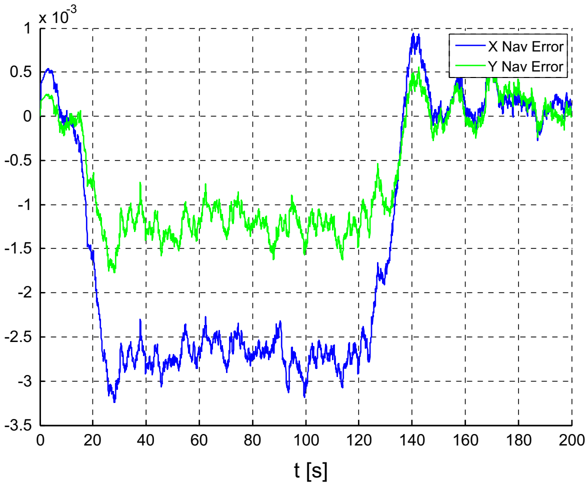


Fig. 20. Kalman filter evaluation based on all sensors (m).

Fig. 17 shows an error equal to 30 mm during absolute navigation and a control outside a dead band of 5.5 mm during relative navigation, matching the mission requirements listed in Fig. 4. During linear maneuver, no actions are taken to control the attitude, since the error remains inside the dead band of 5°, set for absolute navigation and greater than that of relative navigation, set to 1°. During the rotation maneuver, the error between desired and simulated attitude is of the order of 4° (Fig. 18). The estimate of the IKF is presented in Fig. 19.

The initial error is due to overshoot during maneuver along x and y respectively.

When the Failure Detection Identification and Recovery modules of the MMA detect a difference between information from the different sensors above a certain threshold, it sends an alarm vector to the IKF in order to ignore that information, from a failed sensor, in the calculation of the estimated state. Inside the Information Kalman Filter, the value of the observation noise of that sensor will be noticeably increased, through the error covariance matrix *R*, so the filter will exclude the unreliable information. The following figures compare the estimated state of the Information Kalman filter in case of nominal IMU and laser function (Fig. 20) and after laser failure (Fig. 21). Different behavior between *x* and *y* errors is the result of thrusters mixing of velocity and attitude control.

The failure of laser during overall mission produces a loss of accuracy in the estimate of the state equal to 12 mm, which is reflected on the control as a loss of the final accuracy on position of 4 cm (Fig. 22), so the docking cannot be performed according to most stringent mission requirements. This loss in accuracy is mainly due to IMU error drift which cannot be reset by the laser in failure. However during relative navigation, the final estimate of Kalman filter doesn't depend on IMU measurements.

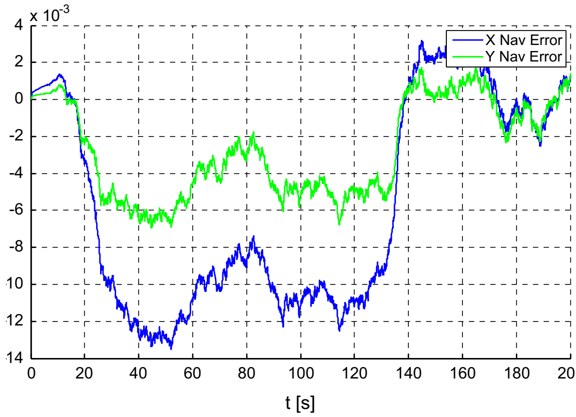


Fig. 21. Kalman filter evaluation without laser sensor (*m*).

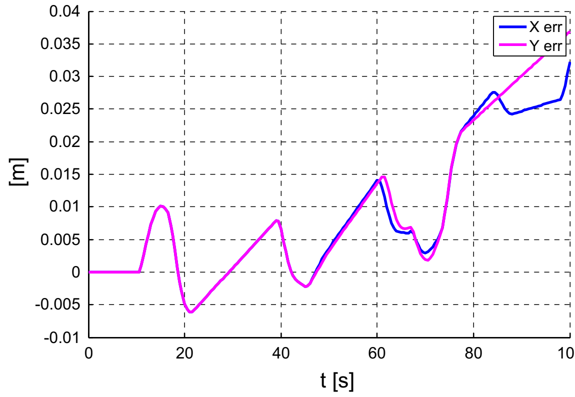


Fig. 22. Position error without laser sensor (absolute reference frame) (m).

7. Conclusions

The GNC algorithms are designed and developed using a simulation environment in which the dynamic model of vehicles including the sensors was considered, implementing the equations of motion between two rigid bodies, one of which, the target, has a known kinematics. The navigation algorithm works both in relative and absolute coordinate frame and estimates the distance at which the switch of reference frame may occur. The chosen Information Kalman filter allows the fusion between multiple sensors and it is able to estimate position, attitude and velocity with a bounded error (i.e. ± 3 mm and 0.5°). The error due to INS integration is reset by the laser measurements. The guidance algorithms calculate the desired velocity and position profile starting from a pre-defined path, assigned inside the MPM, and compare them with the navigation output. The control algorithm manages the activation of 14 cold thrusters and the respective level of thrust. If the laser sensor fails during absolute navigation, the error on position is of the order of 4 cm, so in this case the requirements are not accomplished. The final accuracy of the GNC, in nominal conditions, ensures the completion of the mission with a precision of 5° and 30 mm up to 2 m from the target, and 5.5 mm for lower range.

As a final remark, even if only final experiments on the physical rig, with hardware in the loop, can confirm the ability of the GNC algorithms to comply with the design requirements, the use of simulations provides a comprehensive approach, a method for assessing and tuning the software prototype, before the complete hardware layout is available. Additionally, a description of the developed RV&D experimental rig is here detailed.

Nomenclature

α_i	Acceleration during *i*-th maneuver
$\overline{b_i}$	Observation noise of *i*-th sensor
$\overline{\overline{B_k}}$	Control input matrix
F_c	Force applied to the chaser
F_t	Force applied to the target
$F(s)$	Transfer function of ON–OFF signal
F_{Y1}	Maximum thrust for *Y*1 group
$\overline{\overline{H_k}}$	Observation matrix
I_y	Required pulse along y body axis
I_{Y1}	Required pulse of *Y*1 group
k	Duration of transient

$\overline{\overline{K_i}}$	Kalman gain of *i*-th sensor
K_I	Integrative gain of controller
K_D	Derivative gain of controller
K_P	Proportional gain of controller
$\overline{\overline{P_k}}$	Error covariance matrix
$\overline{\overline{Q_k}}$	Process noise covariance
$\overline{\overline{R_i}}$	Measurement noise covariance of *i*-th sensor
T_{stk}	Duration of station keeping maneuver
$\overline{u}$	Control vector
V_{di}	Drift velocity during *i*-th maneuver
x_i, y_i	Waypoint coordinates of *i*-th maneuver
$\overline{x}$	State vector
x_{gui}, y_{gui}	Desired position in switching reference frame
x_{nav}, y_{nav}	Estimated position in switching reference frame
$\overline{z_i}$	Measurement vector of *i*-th sensor
ΔT_{at}	Duration of attitude maneuver
ΔT_{lin}	Duration of linear maneuver
ΔT_{con}	PID sampling time
ΔT_{PWM}	PWM sampling time
ϑ_j	Angular rotation of *j*-th maneuver
$\dot{\theta}_j$	Angular velocity of *j*-th maneuver
$\ddot{\theta}_j$	Angular acceleration of *j*-th maneuver
τ_{thr}	Percentage of thruster ON-time

Acronyms

CC	Control Center
MPM	Mission Plan matrix
CoG	Center of Gravity
PID	Proportional Integrative Derivative
DOF	Degree of Freedom
PWM	Pulse Width Modulator
GNC	Guidance Navigation Control
RV&D	Rendezvous & Docking
INS	Inertial Navigation System
STEPS	System and Technologies for Space ExPloration
MMA	Mission Management Agent
ZMWG	Zero Mean White Gaussian

Acknowledgments

The authors wish to acknowledge the invaluable technical support given by Mrs Stefania Galli (ALTRAN Italia) and IMEXA for the development of the MMA software.

References

[1] W. Fehse, Automated Rendezvous and Docking for Spacecraft, Cambridge University Press, UK, 2003.

[2] M. Romano, D.A. Friedman, T.J. Shay, Laboratory experimentation of autonomous spacecraft approach and docking to a collaborative target, J. Spacecr. Rockets 44 (No. 1) (2007) 164–173.

[3] W.H. Clohessy, R.S. Whiltshire, Terminal guidance system for satellite rendezvous, J.Aerosp. Sci. 27 (No. 9) (1960) 653–658.

[4] M.E. Polites, An Assessment of the Technology of Automated Rendezvous and Capture in Space, Nasa TP-1998-208528, 1998.

[5] I. Kawano, M. Mokuno, T. Kasai, T. Suzuki, Results of autonomous rendezvous and docking experiments of engineering test satellite-VII, J. Spacecr. Rockets 38 (No. 1) (2001) 105–111.

[6] R. Bevilacqua, M. Romano, F. Curti, A. Caprari, V. Pellegrini, Guidance navigation and control for autonomous spacecraft assembly: analysis and experimentation, Int. J. Aerosp. Eng. 2011 (2011) 1–18.

[7] A. Saenz-Otero, A. Chen, SPHERES: development of an iss laboratory for formation flight and docking research, in: Proceedings of the IEEE Aerospace Conference, Big Sky, USA, 2002.

[8] S. Nolet, The SPHERES Navigation System: from early development to on-orbit testing, in: Proceedings of the AIAA Guidance Navigation and Control Conference, Hilton Head, USA, 2007.

[9] T. Boge, E. Schreutelkamp, A new command control environment for rendezvous and docking simulation at the epos-facility, in: Proceedings of the 7th International Workshop on Simulation for European Space Programs, Nordwijk, The Netherlands, 2002.

[10] 〈http://dssl.technion.ac.il/〉.

[11] M. Wette, G. Sohl, D. Scharf, E. Benowitz, The formation algorithms and simulation test-bed, in: Proceedings of the 2nd International Symposium on Formation Flying Missions and Technologies, Washington D.C., USA, 2004.

[12] F. Caron, E. Duflos, D. Pomorski, P. Vanheeghe, GPS/IMU data fusion using multisensor kalman filtering: introduction to contextual aspects, Inf. Fusion 7 (No. 2) (2006) 221–230.

[13] R.E. Kalman, A New Approach to Linear Filtering and Prediction Problems, J. Basic Eng. 82 (Series D) (1960) 35–45.

[14] S.W. Jeon, S. Jung, Novel limit cycle analysis of the thruster control system with time delay using a pwm-based pd controller, in: Proceedings of the IEEE International Symposium on Industrial Electronics, Seoul, South Korea, 2009.

[15] B. Wie, Space Vehicle Dynamics and Control, American Institute of Aeronautics and Astronautics, Reston, VA, USA, 1998.

[16] O.J. Woodman, An Introduction to Inertial Navigation, University of Cambridge, Cambridge, UK, 2007 TR-696.